Arthur-Merlin Games in Boolean Decision Trees

Ran Raz* Gábor Tardos†
Oleg Verbitsky‡ Nikolai Vereshagin§

Abstract

It is well known that probabilistic boolean decision trees cannot be much more powerful than deterministic ones (N. Nisan, *SIAM Journal on Computing*, 20(6):999–1007, 1991). Motivated by a question if randomization can significantly speed up a nondeterministic computation via a boolean decision tree, we address structural properties of Arthur-Merlin games in this model and prove some lower bounds.

We consider two cases of interest, the first when the length of communication between the players is limited and the second if it is not. While in the first case we can carry over the relations between the corresponding Turing complexity classes, in the second case we observe in contrast with Turing complexity that a one-round Merlin-Arthur protocol is as powerful as a general interactive proof system and, in particular, can simulate a one-round Arthur-Merlin protocol.

Moreover, we show that sometimes a Merlin-Arthur protocol can be more efficient than an Arthur-Merlin protocol, and than a Merlin-Arthur protocol with limited communication. This is the case for a boolean function whose set of zeroes is a code with high minimum distance and a natural uniformity condition.

Such functions provide an example when the Merlin-Arthur complexity is 1 with one-sided error $\epsilon \in (\frac{2}{3}, 1)$, but at the same time the nondeterministic decision tree complexity is $\Omega(n)$. The latter should be contrasted with another fact we prove. Namely, if a function has Merlin-Arthur complexity 1 with one-sided error probability $\epsilon \in (0, \frac{2}{3}]$, then its nondeterministic complexity is bounded by a constant.

Other results of the paper include connections with the block sensitivity and related combinatorial properties of a boolean function.

*Department of Applied Mathematics, Weizmann Institute, Rehovot 76100, Israel.

†Mathematical Institute of the Hungarian Academy of Sciences, Pf. 127, Budapest, H-1364 Hungary.

‡Department of Mechanics and Mathematics, Lviv University, 290602 Lviv, Ukraine. Partially supported by INTAS grant No. 96-0753 and an ÖAD grant. Part of this work was done while visiting the Institute of Information Systems, Vienna University of Technology.

§Department of Mathematical Logic and Theory of Algorithms, Moscow State University, Vorobjevy Gory, 119899 Moscow, Russia.

1 Introduction

A boolean decision tree is an algorithm that computes a boolean function $f(x_1, \dots, x_n)$ by asking, step by step, values of the variables $x_1, \dots, x_n$. Each choice of a variable to ask is based on the knowledge of the variables that have been asked before. The cost of computation is the number of variables to be queried. By $\mathrm{d}(f)$ we denote the minimum number of queries needed for a decision tree to compute f on every input.

Randomized computations via decision trees can be defined in a standard vein. Let us denote the corresponding complexity measure by $\mathrm{r}(f)$. In this section we assume the error probability $\frac{1}{3}$ unless it is specified explicitly. It is well known [12] that randomization cannot help much in boolean decision trees. More specifically,

$$\mathrm{d}(f) = O(\mathrm{r}(f)^3). \tag{1}$$

It is quite natural to ask if randomization can significantly speed up a nondeterministic computation. Two models combining randomness and nondeterminism are suggested in [1] *(Arthur-Merlin games)* and [5] *(interactive proof systems)* and both can be directly extended over boolean decision trees. Our work is motivated by a question (posed in [15]) if these models can be more efficient than a mere nondeterministic decision tree.

First we address structural properties of interactive proof systems and Arthur-Merlin games in boolean decision trees. We consider two cases, the first when the length of communication between the players is limited and the second if it is not. The case when the restriction on communication is a polylogarithm of input size n is of particular interest, since it is closely related to computations via polynomial time Turing machines with access to an oracle (see, e.g., [16, page 294] and [11, Section 5.3] for formal treatment). Bounds on the boolean decision tree complexity are useful tools in constructing oracles with desired relations between Turing complexity classes and in proving conditional results [2, 7, 8].

Conversely, all the facts proven for the corresponding Turing complexity classes that hold true under any oracle can be directly carried over to decision trees. We mention three examples.

1. Arthur-Merlin games are as powerful as a general interactive proof system [6].

2. The error in an Arthur-Merlin game can be made one-sided [18, 4].

3. A one-round Arthur-Merlin game can simulate a one-round Merlin-Arthur game [1].

Let us state the latter fact more accurately. We use the following notation. By $\mathrm{am}(f)$ and $\mathrm{ma}(f)$ we denote the complexity measures in the boolean decision tree model that correspond to one-round Arthur-Merlin and Merlin-Arthur games, respectively. Here we assume no limitations on the length of communication. When we allow Merlin to send messages of length at most l, we supply the corresponding measures with upper index l writing $\mathrm{am}^{(l)}(f)$ and $\mathrm{ma}^{(l)}(f)$. We impose no restrictions on the number of Arthur's random bits (see Remark 2.1 below). Then a formal statement of the above claim 3, that follows from [1], is

$$\mathrm{am}^{(l)}(f) = O(l\,\mathrm{ma}^{(l)}(f)). \tag{2}$$

Other complexity measures we are interested in also have limited and unlimited versions. $\mathrm{ip}^{(l)}(f)$ denotes the complexity measure of a boolean function with respect to a many-round Arthur-Merlin game with total length of Merlin's messages at most l, while the measure $\mathrm{ip}(f)$ is respectively to an interactive proof system without any limitations in the decision tree model. Similarly, $\mathrm{nd}^{(l)}(f)$ refers to the nondeterministic decision tree complexity with witness of length at most l, while $\mathrm{nd}(f)$ is its powerful version. Note that for the one-round Arthur-Merlin game and nondeterministic complexities the limitation on the length of communication is significant only if it is rather strict. More specifically, we have the following implications:

$$\begin{aligned} \mathrm{nd}(f) \le d \;&\Rightarrow\; \mathrm{nd}^{(d\log 2n)}(f) \le d, \\ \mathrm{am}(f) \le d \;&\Rightarrow\; \mathrm{am}^{(d\log 2n)}(f) \le d, \end{aligned}$$

where n stands for the number of variables of f.

It turns out that if we do not restrict the length of communication, the structural properties 1 and 2 can be strengthened and proven much simpler.

1. A one-round Merlin-Arthur game is as powerful as a general interactive proof system, i.e., $\mathrm{ma}(f) = \mathrm{ip}(f)$.

2. Error probability ϵ in a one-round Merlin-Arthur game can be made one-sided at cost of increasing it to $\frac{\epsilon}{1-\epsilon}$.

As for property 3, relation (2) without any limits on l becomes meaningless. Instead, by item 1 we have $\mathrm{ma}(f) \le \mathrm{am}(f)$. Thus, we have two hierarchies of complexity measures

$$\mathrm{ip}^{(l)}(f) \le \mathrm{am}^{(l)}(f), \quad \mathrm{am}^{(l)}(f) = O(l\,\mathrm{ma}^{(l)}(f)), \quad \mathrm{ma}^{(l)}(f) \le \mathrm{nd}^{(l)}(f)$$

(parallel to inclusions $NP \subseteq MA \subseteq AM \subseteq IP$ in Turing complexity), and

$$\mathrm{ip}(f) = \mathrm{ma}(f) \le \mathrm{am}(f) \le \mathrm{nd}(f). \tag{3}$$

The problem is how dense or sparse these hierarchies are.

The main result of this paper shows a large gap between $\mathrm{ma}(f)$ and $\mathrm{nd}(f)$, and some gap even between $\mathrm{ma}(f)$ and $\mathrm{am}(f)$. A large gap is also shown between $\mathrm{ma}(f)$ and $\mathrm{ma}^{(l)}(f)$ for l much smaller than n.

It is useful to get more broad view of the situation by prefixing some lower bounds on $\mathrm{ip}(f)$ to (3). The first bound of interest is

$$\mathrm{ip}(f) > \mathrm{bs}(f)/2, \tag{4}$$

where $\mathrm{bs}(f)$ denotes *the block sensitivity* of a boolean function [12] [1], the maximum number of zeroes of f that differ from some one of f in disjoint blocks of variables. This is a simple extension of the bound $\mathrm{r}(f) = \Omega(\mathrm{bs}(f))$ from [12]. Note that bound (4) together with relations

$$\mathrm{nd}(f) \leq \mathrm{bs}(f)\,\mathrm{bs}(\neg f) \tag{5}$$

and

$$\mathrm{d}(f) \leq \mathrm{nd}(f)\,\mathrm{nd}(\neg f) \tag{6}$$

proven in [12] and [2, 7, 14], respectively, implies the relation $\mathrm{d}(f) = O(\mathrm{ip}(f)^2\,\mathrm{ip}(\neg f)^2)$, which is a qualitative generalization of (1) and (6).

We suggest also a bound that is in a sense tighter. Namely,

$$\mathrm{ip}(f) > \mathrm{sep}(f)/2, \tag{7}$$

where $\mathrm{sep}(f)$ is a combinatorial characteristic of a boolean function that we call *separability of* f and define as follows. Given w, a one of f, and D, a set of zeroes of f, let $\mathrm{sep}(w, D)$ denote the minimum s such that w can be distinguished from any element of D by looking at only s positions. $\mathrm{sep}(f)$ is the maximum of $\frac{\mathrm{sep}(w,D)}{\log |D|}$ over all w and D. It is easy to see that $\mathrm{sep}(f) \geq \frac{\mathrm{bs}(f)}{\log \mathrm{bs}(f)}$. The hierarchy (3) can now be updated to

$$\mathrm{bs}(f)/(2\log \mathrm{bs}(f)) \leq \mathrm{sep}(f)/2 \leq \mathrm{ip}(f) = \mathrm{ma}(f) \leq \mathrm{am}(f) \leq \mathrm{nd}(f).$$

We point out a simple example when $\mathrm{bs}(f) \leq 3$ and at the same time $\mathrm{sep}(f) > \frac{\sqrt{n}}{\log n}$. Thus, (7) can be considered as a sharpening of (4). This example also shows that $\mathrm{bs}(f)$ can be much smaller than $\mathrm{nd}(f)$ (though both $\mathrm{bs}(f)$ and $\mathrm{bs}(\neg f)$ cannot by (5)). We will see that $\mathrm{sep}(f)$ also can sometimes be much smaller than $\mathrm{nd}(f)$. Therefore, (7) cannot help if we try to show that $\mathrm{ip}(f)$ and $\mathrm{nd}(f)$ are polynomially related. But if we are going to prove the opposite, (7) becomes useful. This relation suggests a domain which we should inspect to solve the $\mathrm{ip}(f)$ versus $\mathrm{nd}(f)$ question. The domain consists of functions f with small $\mathrm{sep}(f)$ and large $\mathrm{nd}(f)$.

We consider the following class of functions that meet both of these conditions. Let U be a binary code with minimum distance δn, where $\delta \in (0, \frac{1}{2})$. Let $F(w) = 1$ iff $w \notin U$. We call F the *check function* of U. It is not hard to see that $\mathrm{sep}(F)$ is

[1] In fact, our terminology and notation slightly differ from [12]. Our definition of the block sensitivity corresponds to notion of the block sensitivity *on 1-instances* in [12], where notation $\mathrm{bs}^1(f)$ is used. We suppress the index for notation simplicity.

bounded by a constant. If in addition U is densely dispersed in the boolean cube, which is a natural property of a code, then $\mathrm{nd}(F)$ is big. Taking U random, we get F with $\mathrm{nd}(F) = \Omega(n)$. It turns out, this construction provides an example of a function with large gap not only between $\mathrm{sep}(f)$ and $\mathrm{nd}(f)$, but even between $\mathrm{ma}(f)$ and $\mathrm{nd}(f)$. We observe that $\mathrm{ma}(F) = 1$ with one-sided error probability $1 - \delta/2$.

Making use of the property that U is dispersed in the boolean cube in a sense uniformly, we prove lower bounds $\mathrm{am}(F) = \Omega(\log n)$ and $\mathrm{ma}^{(l)}(F) = \Omega(\frac{n}{l})$. Summing up, we have got an example of boolean function F for which the following bounds are true simultaneously:

- $\mathrm{ma}(F) = O(1)$;
- $\mathrm{nd}(F) = \Omega(n)$;
- $\mathrm{ma}^{(l)}(F) = \Omega(\frac{n}{l})$;
- $\mathrm{am}(F) = \Omega(\log n)$.

The main question we leave open is if the complexity measures $\mathrm{am}(f)$ and $\mathrm{nd}(f)$ are polynomially related. It would be insightful to improve our logarithmic lower bound on $\mathrm{am}(F)$ or, alternatively, give an upper bound.

A related question is if $\mathrm{am}(f) = O(1)$ implies $\mathrm{nd}(f) = O(1)$. In the last part of the paper we prove such an implication in a few particular cases. We here mention only one claim of such a kind. Namely, $\mathrm{ma}(f) \leq 1$ implies $\mathrm{nd}(f) \leq 2$ if the error probability is in range $(0, \frac{2}{5}]$ or the one-sided error is in $(0, \frac{2}{3}]$. This should be contrasted with our example of function F, for which $\mathrm{nd}(F) = \Omega(n)$ and at the same time $\mathrm{ma}(F) = 1$ with one-sided error $1 - \delta/2$. Notice that the error here can be arbitrary in the interval $(\frac{3}{4}, 1)$, as δ can be taken arbitrarily close to $\frac{1}{2}$. Moreover, we are able to improve this range to $(\frac{2}{3}, 1)$, thereby showing that $\frac{2}{3}$ is the exact threshold in such kind of examples.

In [9] the *characteristic* functions of codes were used to obtain lower bounds for some kind of branching programs. It is interesting to note that both [9] and our paper employ in essence the same properties of codes.

The paper is organized as follows. In Section 2 we define the model and complexity measures under consideration and discuss their structural properties. Section 3 establishes relations with the block sensitivity and the separability. In Section 4 we introduce check functions for codes and estimate their complexity. We in detail give a probabilistic construction of codes with the desired properties, mention an algebraic-geometry construction, and discuss what can be shown for some classical codes. In Section 5 we consider properties of functions with Merlin-Arthur complexity bounded by a constant. Section 6 sums up our considerations and lists open questions.

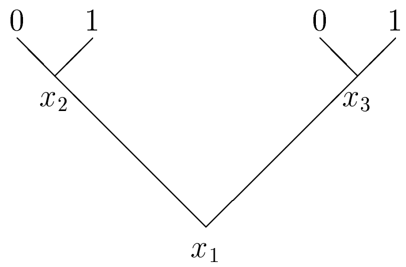

Figure 1. This depth-2 deterministic decision tree computes the boolean function $(1-x_1)x_2+x_1x_3$.

2 The model and its structural properties

A *deterministic boolean decision tree* T over the variable set $X = \{x_1, \dots, x_n\}$ is a rooted, ordered, binary tree. Each internal node has two out-going edges and one in-going (excepting the root). Additionally, each internal node is labelled by a variable from X, and each leaf is labelled by either 0 or 1.

The decision tree T *computes* a boolean function $f(x_1, \dots, x_n)$ in the following sense. Each boolean assignment w to the variables $x_1, \dots, x_n$ determines a path in T from the root to a leaf by the following rule. If the label of an internal node evaluates to 0, we choose the left out-going edge; we choose the right one otherwise. The value of f on w must agree with the label of the leaf at the end of the path (see Figure 1). We write $T(w) = f(w)$.

A *depth-d nondeterministic boolean decision tree* S is a collection of depth-d deterministic decision trees. We write $S(w) = 1$ in the case that at least one deterministic member of S outputs 1 on w; otherwise we write $S(w) = 0$. We say that S computes f if $S(w) = f(w)$ for any assignment w.

Denote the size of set S by $|S|$. We say that tree S is of *nondeterminism* l if l is an integer and $\log |S| \leq l$. This number means the length of a prompt sufficient to find a 1-path consistent with an input.

Equivalently, one can view a nondeterministic tree as several deterministic branches that go out of a common unlabelled root (see Figure 2). As usually in a nondeterministic model, such a tree outputs 1 on input w if there is a path from the root to an 1-leaf that agrees with w. The root is a peculiar *nondeterministic node* that has arbitrary out-degree and is disregarded when counting the depth. Had we allowed many such nondeterministic nodes, the model would not have become more powerful. The tree is of nondeterminism $\lceil \log L \rceil$, where L is the out-degree of the nondeterministic node.

A *depth-d probabilistic decision tree* R is a probability distribution over the set of all the depth-d deterministic decision trees. Suppose that a deterministic decision tree T is taken randomly according to R, and let p be the probability that $T(w) = 1$ for an assignment w. Then we say that $R(w) = 1$ with probability p, and $R(w) = 0$

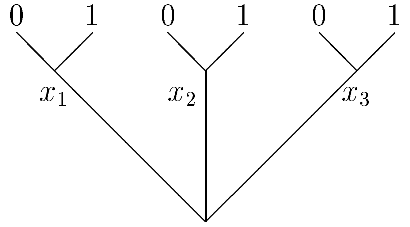

Figure 2. This depth-1 nondeterministic decision tree computes the disjunction of variables x_1, x_2, x_3.

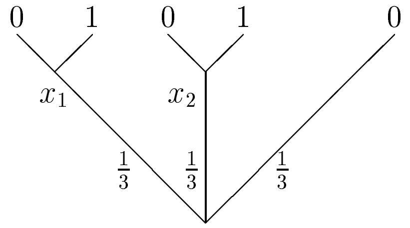

Figure 3. This depth-1 probabilistic decision tree computes the conjunction of variables x_1, x_2 with error $\epsilon > 1/3$.

with probability $1-p$. R computes f *with error* ϵ if for any assignment w, $R(w) = f(w)$ with probability more than $1-\epsilon$.

Remark 2.1 It is not hard to show that if a function $f(x_1, \ldots, x_n)$ is computable via a depth-d probabilistic decision tree with error ϵ, then it can be computed via a depth-d probabilistic decision tree R which is the uniform distribution on set of size $O(n/\delta^2)$ of depth-d deterministic trees, with error $\epsilon + \delta$ for any $\delta < \epsilon(1-\epsilon)$. Similar properties hold true for other probabilistic models below. This is the reason why we measure nondeterminism but pay no attention to *randomness* in the models under consideration.

We can view a probabilistic tree as several deterministic branches that go out of a common root. The root is a *probabilistic node* of arbitrary out-degree that does not contribute to the tree depth. It is unlabelled, but the out-going edges are labelled by probabilities that sum to 1. Any path from the probabilistic node is chosen with the assigned probability (see an example in Figure 3). Had we allowed many such probabilistic nodes, the model would not have become more powerful.

A *depth-d Arthur-Merlin decision tree* Q [of nondeterminism l] is a probability distribution over the set of all the depth-d nondeterministic decision trees [of nondeterminism l]. Q computes f with error ϵ if for any assignment w, $S(w) = f(w)$

with probability more than $1-\epsilon$, where a nondeterministic tree S is taken randomly according to Q. In detail this condition can be rewritten as follows.

1. If $f(w) = 1$, then a random S has a deterministic branch T such that $T(w) = 1$ with probability exceeding $1 - \epsilon$.

2. If $f(w) = 0$, then a random S has a deterministic branch T such that $T(w) = 1$ with probability less than ϵ.

A *depth-d Merlin-Arthur decision tree* Q is a collection of depth-d probabilistic decision trees. Q computes f with error ϵ if for any assignment w the following is true.

1. If $f(w) = 1$, then for some $R \in Q$ we have $R(w) = 1$ with probability exceeding $1 - \epsilon$.

2. If $f(w) = 0$, then for all $R \in Q$ we have $R(w) = 1$ with probability less than ϵ.

If in the first case we have a stronger condition that $R(w) = 1$ with probability 1, we say that Q computes f with *one-sided error* ϵ. We say that Merlin-Arthur decision tree Q is of *nondeterminism* l if l is an integer not less than $\log |Q|$.

Note that the Arthur and Merlin trees admit a visual interpretation using both probabilistic and nondeterministic nodes similarly to probabilistic and nondeterministic trees.

The next model is most general. A *depth-d interactive decision tree* Q is a collection of deterministic depth-d trees $\{T_i\}$ indexed by elements of set I. Given set I, for each assignment w we consider a game of two persons, the verifier and the prover, that proceeds as follows. At the beginning the verifier picks a random string r, unknown to the prover, and initiates the message exchange between the players. In j-th round of the exchange, the verifier sends the prover message a_j, after which the prover sends the verifier message b_j. The choice of a_j by the verifier is determined by $r, b_1, \ldots, b_{j-1}$. The prover's message b_j is a function of $a_1, \ldots, a_{j-1}, a_j$. This function is called a *strategy* of the prover. After some number k of rounds, the verifier terminates the game. Let $i = \langle r, a_1, b_1, \ldots, a_k, b_k \rangle$. The prover wins if $T_i(w) = 1$.

An interactive tree Q computes a function f with error ϵ if the following conditions are met.

1. If $f(w) = 1$, then the prover has a strategy that wins with probability more than $1 - \epsilon$, where the probability is taken over random strings r.

2. If $f(w) = 0$, then the prover wins with probability less than ϵ irrespectively of his strategy.

By $\mathrm{d}(f)$ $[\mathrm{nd}(f)]$ we denote the minimum depth of a deterministic [nondeterministic] decision tree computing a boolean function f. The minimum depth of a probabilistic [Arthur-Merlin, Merlin-Arthur, interactive] decision tree that computes f with error ϵ is denoted by $\mathrm{r}_\epsilon(f)$ $[\mathrm{am}_\epsilon(f), \mathrm{ma}_\epsilon(f), \mathrm{ip}_\epsilon(f)]$. We use notation $\mathrm{ma}_\epsilon^{(l)}(f)$ and $\mathrm{am}_\epsilon^{(l)}(f)$ for the case when nondeterminism is limited by l. $\mathrm{ma}_{0,\epsilon}(f)$ stands for the complexity of computing f by a Merlin-Arthur tree with one-sided error ϵ.

Theorem 2.2 *For any boolean function f and error $\epsilon < 1/2$ we have the following relations:*

(i) $\mathrm{ma}_\epsilon(f) = \mathrm{ip}_\epsilon(f)$;

(ii) $\mathrm{ma}_{0,\frac{\epsilon}{1-\epsilon}}(f) = \mathrm{ma}_\epsilon(f)$;

(iii) $\mathrm{ma}_\epsilon(f) \le \mathrm{am}_\epsilon(f) \le \mathrm{am}_\epsilon^{(l)}(f) \le c_\epsilon l\, \mathrm{ma}_\epsilon^{(l)}(f)$, *where c_ϵ is a constant depending on ϵ.*

Proof: (i) The part "$\ge$" is trivial. We prove the part "$\le$". Let Q be an interactive tree computing f. Fixing prover's strategy converts Q into a probabilistic tree. For each w such that $f(w) = 1$, choose an optimal strategy and denote the corresponding probabilistic tree by R_w. Compose a Merlin-Arthur tree from all R_w rooting them at a nondeterministic node. As easily seen, this tree has the same depth as Q does and computes f with the same error.

(ii) The inequality "$\ge$" is a simple universal relation. Let us prove the inequality "$\le$". Consider a Merlin-Arthur tree Q computing f with error ϵ. We will denote ones of f by w, and zeroes by u. For each one w, tree Q contains a probabilistic branch R_w such that $R_w(w) = 1$ with probability greater than $1-\epsilon$, while $R_w(u) = 1$ with probability smaller than ϵ for all zeroes u. Let R'_w be a distribution induced by R_w on those deterministic branches that evaluate to 1 on input w. Clearly, $R'_w(w) = 1$ with probability 1. For any zero u we have

$$\mathbf{P}\left[R'_w(u) = 1\right] = \mathbf{P}\left[R_w(u) = 1 \,|\, R_w(w) = 1\right] \le \frac{\mathbf{P}\left[R_w(u) = 1\right]}{\mathbf{P}\left[R_w(w) = 1\right]} < \frac{\epsilon}{1-\epsilon}.$$

This means that a Merlin-Arthur tree Q' consisting of probabilistic branches R'_w, for all ones w, computes f with one-sided error $\frac{\epsilon}{1-\epsilon}$.

(iii) The first inequality is an immediate consequence of item (i). The second is trivial. The third is a translation of [1, Theorem 2.1] into our model. ■

We conclude this section with some terminology that will be used throughout the paper. A *d-cylinder* is a subset of $\{0,1\}^n$ obtained by fixing any $d \le n$ boolean components. A set $W \subseteq \{0,1\}^n$ is called *d-open* if it is a union of d-cylinders.

Let W be the set of ones of a boolean function f. It is not hard to see that $\mathrm{nd}(f) \le d$ iff W is d-open. Indeed, if W is recognizable by a nondeterministic tree of depth d, then each 1-path in the tree determines an assignment to d variables whose any extension w belongs to W. Thus, each 1-path determines a d-cylinder within W. Since every string w from W agrees with an 1-path in the tree, W is the union of all these d-cylinders. Conversely, a d-cylinder is obviously recognizable by a deterministic tree of depth d. Gathering such trees together, we obtain a nondeterministic tree for a d-open set.

This observation also shows that depth-d nondeterministic trees are equivalent with d-DNF boolean formulae.

A *d-neighborhood* of an element $w \in \{0,1\}^n$ is a d-cylinder containing w.

3 Block sensitivity and separability

Given $w \in \{0,1\}^n$ and a block of positions $P \subseteq [n]$, we define $w^{(P)} \in \{0,1\}^n$ to be a boolean vector such that w and $w^{(P)}$ differ exactly at positions from P. Given a boolean function $f : \{0,1\}^n \to \{0,1\}$ and $w \in \{0,1\}^n$, by $\mathrm{bs}_w(f)$ we denote the maximum size of a family of disjoint blocks $P_1, \ldots, P_t \subseteq [n]$ such that all the values $f(w^{(P_1)}), \ldots, f(w^{(P_t)})$ differ from $f(w)$. The *block sensitivity* $\mathrm{bs}(f)$ of function f (on 1-instances) is the maximum of $\mathrm{bs}_w(f)$ over all w such that $f(w) = 1$.

We say that a cylinder C *separates* $w \in \{0,1\}^n$ from $D \subseteq \{0,1\}^n$ if C contains w and is disjoint with D. By $\mathrm{sep}(w, D)$ we denote the minimum d such that there is a d-cylinder C separating w from D. We define the *separability* $\mathrm{sep}(f)$ of a boolean function f to be the maximum of $\frac{\mathrm{sep}(w,D)}{\log|D|}$ over all ones w and sets D of zeroes of f with $|D| > 1$.

Bound (ii) in the theorem below is an easy extension of the bound $\mathrm{r}_\epsilon(f) \geq (1-2\epsilon)\,\mathrm{bs}(f)$ in [12]. We include the proof for completeness.

Theorem 3.1 *For any boolean function f we have the following bounds:*

(i) $\mathrm{ma}_{0,\epsilon}(f) > (1-\epsilon)\,\mathrm{bs}(f)$;

(ii) $\mathrm{ip}_\epsilon(f) > \frac{1-2\epsilon}{1-\epsilon}\,\mathrm{bs}(f)$;

(iii) $\mathrm{ma}_{0,\epsilon}(f) > \mathrm{sep}(f)/(1 + (\log\frac{1}{\epsilon})^{-1})$;

(iv) $\mathrm{ip}_\epsilon(f) > \mathrm{sep}(f)/(1 + (\log\frac{1-\epsilon}{\epsilon})^{-1})$.

Proof: (i) Let $\mathrm{ma}_{0,\epsilon}(f) = d$ and Q be a depth-d Merlin-Arthur tree that computes f with one-sided error ϵ. Consider arbitrary w such that $f(w) = 1$. There is a probabilistic branch R of Q such that $R(w) = 1$ with probability 1 and $R(u) = 1$ with probability less than ϵ for any u with $f(u) = 0$. Number deterministic branches of R arbitrarily, say, $T_1, T_2, \ldots$. For each m, $T_m(w) = 1$. Thus, w determines a 1-path in T_m. Fixing all the variables along this path according to w, we get a d-cylinder C_m

such that $w \in C_m$ and all elements of C_m are accepted by T_m. We will view R, which is a probability distribution over deterministic trees $T_1, T_2, \ldots$, as a distribution over their numbers. Let $\mathbf{m}$ be chosen randomly in accordance with R.

Let $P_1, \ldots, P_t \subseteq [n]$ be the largest family of disjoint blocks such that all the elements $w^{(P_1)}, \ldots, w^{(P_t)}$ are zeroes of f. It suffices to show that $d > (1-\epsilon)t$.

By I_m we denote the set of positions, whose entries define cylinder C_m. Define E to be the average number of $j \in [t]$ for which P_j and $I_{\mathbf{m}}$ intersect. Since for any fixed set I_m this number is at most d, we have the inequality $E \leq d$.

Now we bound E from below. By linearity of the mathematical expectation we have $E = \sum_{j=1}^{t} E_j$, where E_j is the probability of P_j intersecting $I_{\mathbf{m}}$. Note that whereas $w \in C_{\mathbf{m}}$ with probability 1, $w^{(S_j)} \in C_{\mathbf{m}}$ with probability less than ϵ. We can conclude that $E_j > 1 - \epsilon$ for all j. This implies $E > (1-\epsilon)t$. Putting together the lower and upper bounds on E, we obtain the desired inequality $d > (1-\epsilon)t$.

(ii) immediately follows from (i) by Theorem 2.2 (i),(ii).

(iii) Consider arbitrary one w and set D of zeroes of f. We use definitions of d-cylinders C_m, where $d = \mathrm{ma}_{0,\epsilon}(f)$, and a random variable $\mathbf{m}$ introduced in the proof of claim (i). Recall that $w \in C_{\mathbf{m}}$ with probability 1, but for any zero u of f we have $u \in C_{\mathbf{m}}$ with probability less than ϵ. Let $\mathbf{u}$ be a random variable distributed over D. It follows that $\mathbf{P}[\mathbf{u} \in C_{\mathbf{m}}] < \epsilon$. This implies that some C_m contains less than an ϵ fraction of D, measured by the distribution of $\mathbf{u}$. We will use this fact for uniform distributions on subsets of D.

For the uniform distribution on D, we have C_{m_1} containing less than an ϵ fraction of D. Considering the uniform distribution on $D \cap C_{m_1}$, we obtain C_{m_2} containing less than an ϵ fraction of $D \cap C_{m_1}$. Next we do the same for $D \cap C_{m_1} \cap C_{m_2}$, obtaining C_{m_3}. Iterating this procedure $t \leq \left\lceil \frac{\log |D|}{\log \frac{1}{\epsilon}} \right\rceil$ times, we get d-cylinders $C_{m_1}, \ldots, C_{m_t}$, whose intersection, which is a dt-cylinder, separates w from D. Thus, $\mathrm{sep}(w, D)/\log|D|$ does not exceed $d \left\lceil \frac{\log |D|}{\log \frac{1}{\epsilon}} \right\rceil / \log |D|$, which is less than $d((\log \frac{1}{\epsilon})^{-1} + 1)$. The claim follows.

(iv) immediately follows from (iii) by Theorem 2.2 (i),(ii). ∎

Theorem 3.2

(i) $\mathrm{sep}(f) \geq \frac{\mathrm{bs}(f)}{\log \mathrm{bs}(f)}$ *whenever* $\mathrm{bs}(f) > 1$.

(ii) *There is a function* $f(x_1, \ldots, x_n)$ *with* $\mathrm{bs}(f) \leq 3$ *and* $\mathrm{sep}(f) > \sqrt{n}/\log n$ *for large* n.

(iii) *There is a function* $f(x_1, \ldots, x_n)$ *with* $\mathrm{bs}(f) \leq 3$ *and* $\mathrm{nd}(f) = \Omega(n)$.

Proposition (i) of the theorem demonstrates that the separability is not less than the block sensitivity upto a logarithmic factor. Moreover, proposition (ii) shows a gap between these values. In this sense, bound (iv) in Theorem 3.1 can be considered as sharpening bound (ii).

Theorem 3.2 (ii) together with Theorem 3.1 (iv) implies that the block sensitivity and the interactive decision tree complexity are polynomially unrelated. Theorem 3.2 (iii) provides even an larger gap between the block sensitivity and the nondeterministic decision tree complexity. Notice that such a gap is impossible between $\mathrm{nd}(f)$ and both of $\mathrm{bs}(f)$ and $\mathrm{bs}(\neg f)$, as $\mathrm{nd}(f) \leq \mathrm{bs}(f)\,\mathrm{bs}(\neg f)$ [12].

Proof: (i) Given a boolean function f, let $\mathrm{bs}(f) = t$. Consider a one w of f such that for a family of disjoint blocks $P_1, \ldots, P_t \subseteq [n]$ the set $D = \{w^{(P_1)}, \ldots, w^{(P_t)}\}$ consists of zeroes of f. Evidently, $\mathrm{sep}(w, D) = t$ and $\mathrm{sep}(f) \geq t/\log t$.

(ii) Define a function $f(x_1, \ldots, x_n)$ by describing its set of zeroes U. Assuming $n = l(l-1)/2$, let $|U| = l$. We construct an l by n matrix M whose rows are elements of U. For every two-element set $\{k_1, k_2\} \subset [l]$, we put into M the column with 0 at positions k_1 and k_2, and 1 elsewhere. All rows of the matrix obtained are distinct, and set U is specified.

To show that $\mathrm{bs}(f)$ is at most 3, consider arbitrary four zeroes $u_1, u_2, u_3, u_4 \in U$. By construction of U, there is a coordinate $i \leq n$ in which u_1 and u_2 have 0, but u_3 and u_4 have 1. This means that there is no w from which u_1, u_2, u_3, u_4 differ in disjoint blocks of positions.

Finally, observe that $\mathrm{sep}(1^n, U) \geq l/2$, as a single position can separate 1^n only from two elements of U. The bound $\mathrm{sep}(f) > \sqrt{n}/\log n$ follows.

(iii) Let $d = \lfloor (\frac{1}{8} \log \frac{8}{5}) n \rfloor$ and $l = \lfloor (\frac{8}{5})^{\frac{n}{4}} \rfloor$. Let the set U of zeroes of f consist of l strings chosen independently and randomly from $\{0,1\}^n$. Denote the set of ones of f by W. Our goal is to show that events $\mathrm{nd}(f) > d$ and $\mathrm{bs}(f) \leq 3$ occur simultaneously with nonzero probability.

Assumption $\mathrm{nd}(f) \leq d$ implies that either 1^n is not included in W or it is included in W together with a d-neighborhood. The first event happens with probability no more than $l/2^n < \frac{1}{3}$. Consider the second possibility.

Let C be a d-neighborhood of 1^n. A u_i does not fall into C with probability $1 - 1/2^d$; therefore, no u_i fall into C with probability $(1 - 1/2^d)^l$. In other words, C is included in W with this probability. As 1^n has no more than n^d d-neighborhoods, W contains a d-neighborhood of 1^n with probability no more than $n^d(1 - 1/2^d)^l \leq \exp(d \ln n - 2^{-d} l) < 1/3$.

Thus, $\mathrm{nd}(f) \leq d$ with probability less than $\frac{2}{3}$.

Similarly to the proof of part (ii), we use the observation that if $\mathrm{bs}(f) > 3$, then there are $u_1, u_2, u_3, u_4 \in U$ that cannot have exactly 2 ones and 2 zeros at one and the same position. It follows that the probability of the event $\mathrm{bs}(f) > 3$ does not exceed $\binom{l}{4}(1 - \binom{4}{2}/2^4)^n < \frac{1}{24} l^4 (5/8)^n \leq \frac{1}{24}$.

Thus, with non-zero probability, both $\mathrm{bs}(f) \leq 3$ and $\mathrm{nd}(f) > d$. ■

In the rest of this section we give lower bounds on Merlin-Arthur and Arthur-Merlin complexities in terms of related complexity measures, whose consideration sometimes can be more preferable. Similarly to Turing complexity, one can consider another acceptance/rejectance criterion for a probabilistic decision tree. By $\mathrm{pp}(f)$ we denote the minimum depth of a probabilistic decision tree R such that for any input w, $f(w)=1$ iff $R(w)=1$ with probability exceeding $\frac{1}{2}$. Equivalently $\mathrm{pp}(f)$ can be characterized as the minimum *order of a perceptron* computing f (see [10] for definitions). One can easy show that $\mathrm{pp}(f) \leq \mathrm{nd}(f)$.

Lemma 3.3 $\mathrm{pp}(f) \leq c_\epsilon l\, \mathrm{ma}_\epsilon^{(l)}(f)$, *where* $c_\epsilon > 1$ *is a constant depending on error* ϵ.

Proof: Consider a depth-d Merlin-Arthur tree Q computing the function f with nondeterminism l. We convert Q into a depth-$c_\epsilon ld$ probabilistic tree R that computes f in the above sense. We first use the standard amplification procedure for each probabilistic branch of Q and decrease the error to 2^{-l-1} at cost of increasing the depth by a $c_\epsilon l$ factor. Second, we make the nondeterministic root of Q probabilistic by assigning probability $1/L$ to every of L out-going edges. At this stage, we get a probabilistic tree R'. Recall that $l = \lceil \log L \rceil$. Now, if $f(w)=1$, then $R'(w)=1$ with probability at least $\frac{1}{L}(1-2^{-l-1}) \geq 2^{-l}(1-2^{-l-1}) > 2^{-l-1}$; while if $f(u)=0$, then $R'(u)=1$ with probability less than 2^{-l-1}.

At the final stage of construction of R we should lift the threshold 2^{-l-1} to $\frac{1}{2}$. For this purpose, with probability $p = (2^l-1)/(2^{l+1}-1)$ tree R immediately outputs 1, and with probability $1-p$ runs tree R'. ■

The last proposition we prove in this section is a lower bound on the Arthur-Merlin complexity which also can be viewed as an alternative characterization thereof. Denote the sets of ones and zeroes of a boolean function f by W and U, respectively. We will consider arbitrary independent random variables $\mathbf{w}$ and $\mathbf{u}$ distributed on W and U. We define the *partial separability* of a function f *with gap* $1-2\epsilon$, where $\epsilon < \frac{1}{2}$, to be the minimum d such that for any random variables $\mathbf{w}$ and $\mathbf{u}$ there is a depth-d nondeterministic tree S for which $\mathbf{E}[S(\mathbf{w})] - \mathbf{E}[S(\mathbf{u})] > 1-2\epsilon$. We denote this characteristic of f by $\mathrm{ps}_\epsilon(f)$. The next lemma is a particular case of the universal observation by Yao [17].

Lemma 3.4 $\mathrm{ps}_\epsilon(f) \leq \mathrm{am}_\epsilon(f) \leq \mathrm{ps}_{\epsilon/2}(f)$.

Proof: Given a boolean function f and a natural number d, consider the following matrix. Rows are indexed by all the depth-d nondeterministic trees S. Columns are indexed by all the pairs $w\#u$, where $w \in W$ and $u \in U$. An entry at the intersection of the row and the column is $S(w) - S(u)$. By $\mathbf{S}$ and $\mathbf{w}\#\mathbf{u}$ we will denote arbitrary random variables distributed over the index sets of the matrix. Applying the min-max theorem for the two person zero sum game determined by this matrix, we obtain the equality

$$\max_{\mathbf{S}} \min_{w\#u} \mathbf{E}\left[\mathbf{S}(w) - \mathbf{S}(u)\right] = \min_{\mathbf{w}\#\mathbf{u}} \max_{S} \mathbf{E}\left[S(\mathbf{w}) - S(\mathbf{u})\right],$$

where the random variables $\mathbf{w}$ and $\mathbf{u}$ in the right hand side are projections of $\mathbf{w}\#\mathbf{u}$. Using linearity of the mathematical expectation, we rewrite this equality in the form

$$\max_{\mathbf{S}} \min_{w\#u} \left(\mathbf{E}\left[\mathbf{S}(w)\right] - \mathbf{E}\left[\mathbf{S}(u)\right]\right) = \min_{\mathbf{w},\mathbf{u}} \max_{S} \left(\mathbf{E}\left[S(\mathbf{w})\right] - \mathbf{E}\left[S(\mathbf{u})\right]\right),$$

where $\mathbf{w}$ and $\mathbf{u}$ in the right hand side can now be considered as arbitrary independent random variables distributed over W and U, respectively.

Consider the inequality $\mathrm{ps}_\epsilon(f) \leq d$. It means that for any random variables $\mathbf{w}$ and $\mathbf{u}$ there is a depth-d nondeterministic tree S with $\mathbf{E}\left[S(\mathbf{w})\right] - \mathbf{E}\left[S(\mathbf{u})\right] > 1 - 2\epsilon$. By the above equality, this is equivalent to the following claim. There exists a probability distribution $\mathbf{S}$ over depth-d nondeterministic decision trees such that for any $w \in W$ and $u \in U$ it is true $\mathbf{E}\left[\mathbf{S}(w)\right] - \mathbf{E}\left[\mathbf{S}(u)\right] > 1 - 2\epsilon$. We can view $\mathbf{S}$ as a depth-d Arthur-Merlin decision tree. Clearly, the latter condition follows from the assumption that $\mathrm{am}_\epsilon(f) \leq d$ and implies that $\mathrm{am}_{2\epsilon}(f) \leq d$. This proves the first and the second inequalities of the lemma. ∎

4 Complexity of the check function for a binary code

In this section we prove the main result of the paper.

Theorem 4.1 *There is a boolean function $F : \{0,1\}^n \to \{0,1\}$ with the following conditions true for any $\epsilon < \frac{1}{2}$, $l \leq n$, and n large enough:*

(i) $\mathrm{ma}_\epsilon(F) = O(1)$;

(ii) $\mathrm{nd}(F) = \Omega(n)$;

(iii) $\mathrm{ma}_\epsilon^{(l)}(F) = \Omega(\frac{n}{l})$;

(iv) $\mathrm{am}_\epsilon(F) = \Omega(\log n)$.

Thus, Theorem 4.1 shows a large gap between the Merlin-Arthur complexity and the nondeterministic complexity. By Theorem 3.1 (iv) this implies that the separability of a boolean function and its nondeterministic complexity are unrelated, improving Theorem 3.2 (iii). A large gap is proven also between the Merlin-Arthur complexity without any restrictions on nondeterminism and that with such restrictions. Finally, a constant versus logarithm gap is established between the Merlin-Arthur and Arthur-Merlin complexity measures.

When seeking for an appropriate function F to meet the claims of Theorem 4.1, we find insightful Theorem 3.1. It suggests to examine functions with low separability. We address one class of such functions, namely, those whose set of zeroes is a binary code with some natural properties. More exactly, we need the following two properties for a code $U \subset \{0,1\}^n$.

Linear minimum distance. Any two codewords of U differ in at least δn positions for some $\delta \in (0, \frac{1}{2})$.

Uniformity. We call a code U *s-uniform* if for any t-cylinder C, where $t \leq s$, a fraction of codewords in U that belong to C is equal to 2^{-t}, i.e., $\frac{|U \cap C|}{|U|} = 2^{-t}$. As easily seen, it is enough to require this condition only for $t = \lfloor s \rfloor$. We will need $s = \alpha n$ for some $\alpha \in (0, \frac{1}{2})$.

We postpone construction of a code with both properties to the end of this section. Note that a linear code is s-uniform iff the minimum distance of its dual exceeds s (see Lemma 4.8 below).

Given a binary code U, we call a boolean function with zeroes exactly in U the *check function* of the code. All of four claims of Theorem 4.1 are true for F being the check function of an αn-uniform code with minimum distance δn, where α and δ are any constants in $(0, \frac{1}{2})$. Each claim directly follows from one of four forthcoming lemmas.

Lemma 4.2 *Let F be the check function of a code $U \subset \{0,1\}^n$ with minimum distance more than δn. Then* $\mathrm{ma}_{0,1-\delta/2}(F) = 1$.

Notice that one-sided error ϵ can be amplified to ϵ^k at cost of increasing the depth by a k factor.

Proof: A Merlin-Arthur tree R we suggest for F consists of probabilistic branches R_w for each w outside U. Denote (one of) the nearest to w codewords by u', and suppose that Hamming distance between w and u' is τn.

First consider the case that $\tau < \delta/2$. Notice that then u' is unique. Let $P \subset \{x_1, \ldots, x_n\}$ be the set of τn variables to which w and u' assign different values. We construct R_w as follows. With probability p to be specified below this probabilistic branch asks a random variable from $\{x_1, \ldots, x_n\}$, and with probability $q = 1 - p$ it asks a random variable from P. R_w accepts iff the answer is consistent with w. Clearly, $R_w(w) = 1$ with probability 1. Also, $R_w(u') = 1$ with probability $p(1 - \tau)$. Notice that any other codeword u lies at distance at least $(\delta - \tau)n$ from w, where δn is the minimun distance of U. So, $R_w(u) = 1$ with probability at most $q + p(1 - \delta + \tau)$. Thus, R_w errs with probability at most $\max\{p(1-\tau), 1 + p(\tau - \delta)\}$. To minimize it, we set $p = 1/(1 + \delta - 2\tau)$ and obtain R_w accepting any codeword with probability at most $\frac{1-\tau}{1+\delta-2\tau}$, which is less than $1 - \delta/2$ for all $\tau < \delta/2$.

Consider the second case that $\tau \geq \delta/2$. Now let R_w ask just a random variable in $\{x_1, \ldots, x_n\}$ and accept iff its value is consistent with w. Obviously, R_w accepts an arbitrary codeword $u \in U$ with probability at most $1 - \tau \leq 1 - \delta/2$. This completes the construction of R and proves the lemma. ∎

Lemma 4.3 *Let F be the check function of an s-uniform code U. Then* $\mathrm{nd}(F) > s$.

Proof: s-uniformity of U means that every s-cylinder contains a zero of F. Therefore, the set of ones of F cannot be s-open. The lemma follows from the discussion concluding Section 2. ∎

Lemma 4.4 *Let F be the check function of an s-uniform code U. Then*

(i) $\mathrm{pp}(F) > s$;

(ii) $\mathrm{ma}_\epsilon^{(l)}(F) > s/(c_\epsilon l)$, *where $c_\epsilon > 1$ is a constant depending on the error ϵ.*

Proof: (i) Assume, to the contrary, that $\mathrm{pp}(F) \leq s$. This means that some depth-s probabilistic tree R accepts any codeword in U with probability at most $1/2$, while any word outside U with probability strictly more than $1/2$. It follows that $\mathbf{E}[R(\mathbf{w})] - \mathbf{E}[R(\mathbf{u})] > 0$, where $\mathbf{w}$ and $\mathbf{u}$ are uniformly distributed on $W = \{0,1\}^n \setminus U$ and U, respectively, and expectation is over distributions $\mathbf{w}$, $\mathbf{u}$, and R. This inequality implies that $\mathbf{E}[T(\mathbf{w})] - \mathbf{E}[T(\mathbf{u})] > 0$ for at least one deterministic branch T of R. Contradictory with this, we show that

$$\mathbf{E}[T(\mathbf{w})] - \mathbf{E}[T(\mathbf{u})] = 0 \tag{8}$$

for any depth-s deterministic decision tree T.

Let χ_C denote the characteristic function of set C. We can write $T(w) = \sum_C \chi_C(w)$, where the sum is over all cylinders C corresponding to 1-paths in T. By linearity of mathematical expectation,

$$\mathbf{E}[T(\mathbf{w})] - \mathbf{E}[T(\mathbf{u})] = \sum_C (\mathbf{E}[\chi_C(\mathbf{w})] - \mathbf{E}[\chi_C(\mathbf{u})]).$$

But if C is a t-cylinder, where $t \leq s$, then by s-uniformity of U we have $\mathbf{E}[\chi_C(\mathbf{w})] = \mathbf{E}[\chi_C(\mathbf{u})] = 2^{-t}$. Equation (8) follows.

(ii) follows from item (i) by Lemma 3.3. ∎

In the next lemma we use the notion of partial separability introduced at the end of Section 3.

Lemma 4.5 *Let F be the check function of an s-uniform code U with minimum distance at least 3 and $s \geq \frac{n \ln \log n}{\log n}$. Then*

$$\mathrm{ps}_\epsilon(F) > \log n - 2 \log\log n$$

for any $\epsilon \in (0,1)$ and sufficiently large n.

Together with Lemma 3.4 this immediately provides a lower bound on $\mathrm{am}_\epsilon(F)$.

Proof: We present two probability distributions on ones and zeroes of F that are indistinguishable by a depth-d nondeterministic decision tree for $d = \lceil \log n - 2\log\log n \rceil$. More specifically, let a random variable $\mathbf{u}$ be uniformly distributed on the entire set U and $\mathbf{w}$ be uniformly distributed on set $W = \left\{ u^{(i)} : u \in U, i \leq n \right\}$, where $u^{(i)}$ is defined to be a string that differs from u exactly at i-th position. As the code distance of U is at least 3, all the $u^{(i)}$ are distinct ones of F. Consider an arbitrary depth-d nondeterministic tree S and denote $p_1 = \mathbf{E}[S(\mathbf{w})]$ and $p_0 = \mathbf{E}[S(\mathbf{u})]$. Our goal is to show that

$$p_1 - p_0 = O\left(\frac{\log\log n}{\log n}\right), \tag{9}$$

which will imply $\mathrm{ps}_\epsilon(F) > d$ for any constant ϵ.

We split U into two parts U_1 and U_0 putting an element u into the first part if $S(u) = 1$ and into the second otherwise. Given u, let us denote the number of i such that $S(u^{(i)}) = 1$ by $m(u)$. Let $m = \mathbf{E}[m(\mathbf{u}) \,|\, \mathbf{u} \in U_0]$.

We claim that

$$p_1 - p_0 \leq \frac{m}{n}\frac{|U_0|}{|U|}. \tag{10}$$

Indeed,

$$\begin{aligned} p_1 - p_0 &= \mathbf{P}\left[S(\mathbf{u}^{(\mathbf{i})}) = 1\right] - \mathbf{P}[S(\mathbf{u}) = 1] \\ &\leq \mathbf{P}\left[S(\mathbf{u}^{(\mathbf{i})} = 1), S(\mathbf{u}) = 0\right] \\ &= \mathbf{P}\left[S(\mathbf{u}^{(\mathbf{i})}) = 1 \,|\, S(\mathbf{u}) = 0\right] \mathbf{P}[S(\mathbf{u}) = 0] \\ &= \frac{m}{n}\frac{|U_0|}{|U|}. \end{aligned}$$

If $\frac{m}{n} \leq \frac{2\ln\log n}{\log n}$, then (9) follows from (10). It remains to prove (9) in the case that $\frac{m}{n} > \frac{2\ln\log n}{\log n}$.

Define $U' = \{u \in U_0 : m(u) > m/2\}$. One can easily check that $\frac{|U'|}{|U_0|} > \frac{m}{2n}$. Together with (10) this gives

$$p_1 - p_0 < 2\frac{|U'|}{|U|}. \tag{11}$$

Next what we do is upper bounding of $|U'|/|U|$. Let $r = \frac{n\ln\log n}{\log n}$ and $t = \lfloor r/d \rfloor$. We now describe a procedure consisting of t steps. In j-th step we build a covering of U' by at most $(2^d - 1)^j$ disjoint dj-cylinders. The initial covering is the entire boolean cube, that is, it consists of one 0-cylinder. Suppose that before j-th step we have a covering of U' by at most $(2^d - 1)^{j-1}$ disjoint $d(j-1)$-cylinders. In j-th step, every $d(j-1)$-cylinder C from the covering should be split into 2^d disjoint dj-cylinders so that at least one of those can be deleted. To do so, we choose an element u in C that belongs to U'. Let P be the set of positions specifying C. We next choose a position $i \notin P$ so that $S(u^{(i)}) = 1$. This can be done as $u \in U'$ and P contains $d(j-1) \le r < m/2$ positions. We split C into sub-cylinders by assigning all the possible values to the variables that are outside P and are queried by S along a path accepting $u^{(i)}$. If the number of such variables is less than d, we assign also arbitrary additional variables. At least one sub-cylinder from the splitting of C does not intersect U' and even U_0, namely, one that contains $u^{(i)}$. The reason is that each element of this sub-cylinder fits the same 1-path of S as $u^{(i)}$ does.

After t steps of the above procedure, we obtain a covering of $|U'|$ by at most $(2^d - 1)^t$ disjoint dt-cylinders. Since $dt \le \frac{n \ln \log n}{\log n} \le s$, we can employ s-uniformity of U. Below the summation goes over all the cylinders C from the covering.

$$\frac{|U'|}{|U|} = \sum_C \frac{|U' \cap C|}{|U|} \le \sum_C \frac{|U \cap C|}{|U|} \le (2^d - 1)^t 2^{-dt}.$$

By (11), we have $p_1 - p_0 < 2(1 - 2^{-d})^t \le 2\exp\{-t2^{-d}\}$. Substituting $d = \lceil \log n - 2 \log\log n \rceil$ and $t = \lfloor r/d \rfloor$, we obtain

$$p_1 - p_0 = O\left(\frac{1}{\log n}\right),$$

and (9) follows. ∎

To complete the proof of Theorem 4.1, it remains to construct an αn-uniform code with minimum distance δn for some constants α and δ in interval $(0,1)$. It is convenient to fix our attention on linear codes, i.e., suppose that U is a linear subspace of $\mathrm{GF}(2)^n$ (see Remark 4.10, though). We first prove that the desired linear code exists by the probabilistic method, then refer to an algebraic-geometry construction, and finally discuss what can be done with use of some classical codes.

Probabilistic construction

We use the Chernoff bound [3] stated in the following form.

Lemma 4.6 *Let $\xi_1, \xi_2, \ldots, \xi_n$ be independent identically distributed random variables taking two values 0 and 1, either with probability $1/2$. Then for any $\delta \in (0, 1/2]$*

$$\mathbf{P}\left[\sum_{i=1}^{n} \xi_i \le \delta n\right] \le 2^{(H(\delta)-1)n},$$

where $H(\delta) = -\delta \log_2 \delta - (1-\delta)\log_2(1-\delta)$. ∎

Lemma 4.7 *If $0 < \delta < 1/2$ and $\beta < 1 - H(\delta)$, then a random $\lfloor \beta n \rfloor$-dimensional code has minimum distance at least δn with overwhelming probability (i.e., with probability $1 - o(1)$ for $n \to \infty$).*

Proof: Denote $k = \lfloor \beta n \rfloor$ Suppose that vectors $X_1, \ldots, X_k$ are chosen in $\mathrm{GF}(2)^n$ randomly and independently (they may happen to be linearly dependent). Denote the subspace spanned by $X_1, \ldots, X_k$ by U. Let us estimate the probability that code U has minimum distance less than δn. Recall that the minimum distance of a linear code is equal to the minimum weight of a non-zero codeword. Consider a linear combination $X = \mu_1 X_1 \oplus \ldots \oplus \mu_k X_k$ with coefficients $\mu_1, \ldots, \mu_k \in \mathrm{GF}(2)$. If at least one of the coefficients is non-zero, then X is uniformly distributed over $\mathrm{GF}(2)^n$. If $X = \xi_1 \ldots \xi_n$, its weight is equal to $\xi_1 + \ldots + \xi_n$. By Lemma 4.6 this is less than δn with probability at most $2^{(H(\delta)-1)n}$. Therefore, U contains a non-zero vector of weight less than δn with probability at most $2^k 2^{(H(\delta)-1)n} \le 2^{(\beta+H(\delta)-1)n}$.

Estimate now the probability of the same event under the condition that $X_1, \ldots, X_k$ are linearly independent. Note that then U is uniformly distributed over all k-dimensional subspaces. Observe that random and independent $X_1, \ldots, X_k$ are linearly independent with probability

$$\begin{aligned}\frac{(2^n-1)(2^n-2)\cdots(2^n-2^{k-1})}{2^{nk}} &= (1-\frac{1}{2^n})(1-\frac{1}{2^{n-1}})\cdots(1-\frac{1}{2^{n-k+1}}) > \\ 4^{-\frac{1}{2^n}-\frac{1}{2^{n-1}}-\cdots-\frac{1}{2^{n-k+1}}} &> \frac{1}{4}.\end{aligned}$$

It follows that a random k-dimensional U has has minimum distance less than δn with probability at most $4 \cdot 2^{(\beta+H(\delta)-1)n}$, which approaches 0 with n increasing. ■

Lemma 4.7 shows that there is no problem with achieving high minimum distance. It suffices to take at random a code of appropriate dimension. To proceed with the uniformity property, we need some preliminaries from linear algebra.

Given $X = x_1 \ldots x_n$ and $Y = y_1 \ldots y_n$ in $\mathrm{GF}(2)^n$, let $\langle X, Y \rangle = x_1 y_1 \oplus \ldots \oplus x_n y_n$ be their inner product. X and Y are called orthogonal if $\langle X, Y \rangle = 0$. The dual code of U is denoted by $U^\perp$. It consists of all those strings that are orthogonal with each codeword of U. It is well known that $\dim U^\perp = n - \dim U$ and that $(U^\perp)^\perp = U$. We refer to the following fact (see e.g. [9]).

Lemma 4.8 *A linear code U is s-uniform iff the minimum distance of $U^\perp$ exceeds s.*

Proof: Denote $t = \lfloor s \rfloor$. Given a set $T \subseteq [n]$ of t coordinates, consider a linear transformation $P_T : U \to \mathrm{GF}(2)^t$ which is the projection onto T. For each $v \in \mathrm{GF}(2)^t$, the set $P_T^{-1}(v)$ is exactly the intersection of U and the t-cylinder C_v specified by assigning v to T.

First observe that U is s-uniform iff $P_T(U) = \mathrm{GF}(2)^t$ for any T. Indeed, if $P_T(U)$ is a proper subspace of $\mathrm{GF}(2)^t$, then the uniformity condition is violated because C_v with $v \notin P_T(U)$ does not intersect U. Conversely, $P_T(U) = \mathrm{GF}(2)^t$ implies that all intersections $C_v \cap U = P_T^{-1}(v)$ are non-empty and, therefore, contain the same number of elements. As they cover U, the uniformity condition follows.

Now show that the inequality $P_T(U) \neq \mathrm{GF}(2)^t$ is true for some T iff the minimum distance of $U^\perp$ does not exceed t, that is, $U^\perp$ contains a vector of weight at most t. Indeed, $P_T(U) \neq \mathrm{GF}(2)^t$ iff all x in U satisfy relation $\langle x, y\rangle = 0$ for some non-zero y whose non-zero coordinates all are in T. It remains to notice that such a y belongs to $U^\perp$ and its weight does not exceed t. The lemma follows. ∎

Thus, we need a linear code $U \subset \mathrm{GF}(2)^n$ such that both U and $U^\perp$ have minimum distances linear in n.

Lemma 4.9 *Let $0 < \alpha, \delta < \frac{1}{2}$ and $H(\alpha) < 1 - H(\delta)$. Then for n sufficiently large, there exists an αn-uniform code with minimum distance at least δn.*

Proof: Let $\alpha' = \alpha + \frac{1}{n}$ and pick β between $H(\alpha')$ and $1 - H(\delta)$. Denote $k = \lfloor \beta n \rfloor$. Take at random a k-dimensional linear code U. By Lemma 4.7 its minimum distance is at least δn with overwhelming probability.

Notice that $U^\perp$ is a random $(n-k)$-dimensional code. Let $n - k = \gamma n$. For n sufficiently large, γ is arbitrarily close to $1-\beta$, so $\gamma < 1 - H(\alpha')$. Once again referring to Lemma 4.7, we have that the minimum distance of $U^\perp$ is at least $\alpha' n = \alpha n + 1$ with overwhelming probability. By Lemma 4.8, U is αn-uniform with the same probability.

As with non-zero probability U is simultaneously αn-uniform and has minimum distance at least δn, we conclude that there exists a code with both these properties. ∎

Remark 4.10 If we take randomly and independently $2^{\beta n}$ words in $\{0,1\}^n$, with high probability we obtain a code with minimum distance at least δn, provided $\delta < \frac{1}{2}$ and $\beta < (\frac{1}{2} - \delta)^2$. One cannot expect that such a code is αn-uniform for a constant α, but with high probability it is *almost αn-uniform* in the following sense: for any t-cylinder C with $t \leq \alpha n$, a fraction of codewords in U that belong to C deviates from 2^{-t} in at most $2^{-2(\alpha n - 1)}$.

The check function of an almost αn-uniform code with minimum distance δn satisfies all the bounds in Theorem 4.1. Though almost uniformity does not suffice to prove item (i) of Lemma 4.4, it suffices to keep item (ii) of this lemma true.

Algebraic-geometry construction

Another way to obtain an αn-uniform code with minimum distance δn is to use the self-dual codes constructed in [13] from algebraic curves. In particular, the construction in [13] gives us a self-dual code over alphabet $\mathrm{GF}(64)$ with minimum distance $0.3n$. Replacing elements of $\mathrm{GF}(64)$ by strings from $\mathrm{GF}(2)^6$ we get a binary code that is $0.05n$-uniform and has minimum distance $0.05n$ (the code length has increased by 6). Both properties hold true for an arbitrary one-to-one replacement, even if the binary code obtained is not linear.

Classical constructions

Somewhat weaker versions of bounds (i)–(iii) (except (iv)) of Theorem 4.1 can be obtained for the check functions of some classical codes. In this subsection we do this for two well-known codes. One of them, namely, the dual of BCH-code, was used in [9], where lower bounds where proven for some kind of branching programs computing characteristic functions of codes. It is interesting to note that both [9] and our paper employ in essence the same properties of codes.

The first code U_1 we consider is the simplest version of the Reed-Solomon code. Let $n = p2^p$ and interpret the boolean cube $\{0,1\}^n$ as the set of functions from $GF(2^p)$ into itself (represented by their graphs). Then U_1 consists of graphs of univariate polynomials over $GF(2^p)$ of degree at most r. It is not hard to check that U_1 is r-uniform and its minimum distance is at least $2^p - r$. We set $r = 2^{p-1}$. Let F_1 be the check function of U_1. Then by Lemmas 4.2, 4.3, and 4.4 we have

- $\mathrm{ma}_\epsilon(F_1) = O(\log n)$;
- $\mathrm{nd}(F_1) = \Omega(\frac{n}{\log n})$;
- $\mathrm{pp}(F_1) = \Omega(\frac{n}{\log n})$ and $\mathrm{ma}_\epsilon^{(l)}(F_1) = \Omega(\frac{n}{l \log n})$.

For the next example, let $U_2 \subseteq \{0,1\}^n$ be the dual of the Bose-Chaudhuri-Hocquenghem code of designed distance $2t+1$. Following [9], we take $t = \lceil \sqrt{n}/4 \rceil$ to ensure $\Omega(\sqrt{n})$-uniformity and minimum distance $\Omega(n)$. For F_2 the check function of U_2, we obtain

- $\mathrm{ma}_\epsilon(F_2) = O(1)$;
- $\mathrm{nd}(F_2) = \Omega(\sqrt{n})$;
- $\mathrm{pp}(F_2) = \Omega(\sqrt{n})$ and $\mathrm{ma}_\epsilon^{(l)}(F_2) = \Omega(\frac{\sqrt{n}}{l})$.

5 Limited Merlin-Arthur vs. limited nondeterministic complexity

Theorem 4.1 leaves open an intriguing question if Arthur-Merlin and nondeterministic complexities are polynomially related. A weak version of this question is if $\mathrm{am}_\epsilon(f) = O(1)$ implies $\mathrm{nd}(f) = O(1)$. We can answer it in affirmative only in the first particular case that $\mathrm{am}_\epsilon(f) = 1$, for all $\epsilon \in (0,1)$.

Theorem 5.1 $\mathrm{am}_\epsilon(f) \leq 1$ *implies* $\mathrm{nd}(f) < \frac{1}{1-2\epsilon}$.

Proof: By Lemma 3.4, it suffices to show that $\mathrm{ps}_\epsilon(f) \leq 1$ implies $\mathrm{nd}(f) < \frac{1}{1-2\epsilon}$.

Consider an arbitrary boolean function $f : \{0,1\}^n \to \{0,1\}$ with $\mathrm{ps}_\epsilon(f) \leq 1$. Denote by W and U the sets of ones and zeroes of f, respectively. Let $d = \mathrm{nd}(f)$. So, set W is d-open and is not $(d-1)$-open. Therefore, there must be a d-cylinder $C \subseteq W$ that is not included into any $(d-1)$-cylinder inside W. Without loss of generality, assume that C is specified by assigning the first d variables to $a \in \{0,1\}^d$. By $a^{(i)} \in \{0,1\}^d$, $i \leq d$, we denote a string that differs from a exactly at i-th position. For any $i \leq d$, there exists a $b_i \in \{0,1\}^{n-i}$ such that $a^{(i)}b_i \in U$. Let random variables $\mathbf{w}$ and $\mathbf{u}$ be uniformly distributed on sets $\{ab_1, \ldots, ab_d\} \subseteq W$ and $\{a^{(1)}b_1, \ldots, a^{(d)}b_d\} \subseteq U$, respectively.

Take an arbitrary depth-1 nondeterministic decision tree S. Denote $p_1 = \mathbf{E}[S(\mathbf{w})]$ and $p_0 = \mathbf{E}[S(\mathbf{u})]$. Let us show that $p_1 - p_0 \leq \frac{1}{d}$. The first case we consider is that some deterministic branch of S asks one of the first d variables and accepts if it has the same value as in a. Then $p_1 = 1$ and $p_0 \geq 1 - \frac{1}{d}$. In the second, opposite case, we observe that $S(a^{(i)}b_i) = 1$ whenever $S(ab_i) = 1$ and , therefore, $p_0 \geq p_1$.

The condition $\mathrm{ps}_\epsilon(f) \leq 1$ means that $p_1 - p_0 > 1 - 2\epsilon$ for some depth-1 nondeterministic tree S. The estimate $d < \frac{1}{1-2\epsilon}$ follows. ∎

Theorem 5.2

(i) $\mathrm{ma}_{0,1/2}(f) \leq 1$ *(or, equivalently,* $\mathrm{ma}_{1/3}(f) \leq 1$*) implies* $\mathrm{nd}(f) \leq 1$*;*

(ii) $\mathrm{ma}_{0,2/3}(f) \leq 1$ *(or, equivalently,* $\mathrm{ma}_{2/5}(f) \leq 1$*) implies* $\mathrm{nd}(f) \leq 2$*;*

(iii) $\mathrm{ma}_{0,1/3}(f) \leq 2$ *(or, equivalently,* $\mathrm{ma}_{1/4}(f) \leq 2$*) implies* $\mathrm{nd}(f) \leq 2$.

The proof is deferred to the end of this section. In comparison with Theorem 5.1, Theorem 5.2 relaxes the premise $\mathrm{am}_\epsilon(f) \leq 1$ to $\mathrm{ma}_\epsilon(f) \leq 1$ and even to $\mathrm{ma}_\epsilon(f) \leq 2$ but only for a restricted range of the error ϵ. Such an improvement cannot be done for all $\epsilon \in (0,1)$, because this will contradict the example given in Section 4. Notice that parameter δ in Lemma 4.9 can be chosen arbitrarily close to $\frac{1}{2}$. Thus, for any $\epsilon > \frac{3}{4}$ this lemma provides a function F for which $\mathrm{ma}_{0,\epsilon}(F) = 1$ by Lemma 4.2 but $\mathrm{nd}(F) = \Omega(n)$ by Lemma 4.3. In fact, we are able to improve this example attaining the error ϵ as small as it is possible in view of Theorem 5.2 (ii).

Theorem 5.3 *For any $\sigma > 0$ there is a boolean function $F : \{0,1\}^n \to \{0,1\}$ such that* $\mathrm{ma}_{0,2/3+\sigma}(F) = 1$ *and* $\mathrm{nd}(F) = \Omega(n)$.

Thus, the value $\epsilon = \frac{2}{3}$ is the exact theshold: if $\epsilon \leq \frac{2}{3}$, then $\mathrm{ma}_{0,\epsilon}(f) \leq 1$ implies $\mathrm{nd}(f) = O(1)$, while if $\epsilon > \frac{2}{3}$, then $\mathrm{ma}_{0,\epsilon}(F) = 1$ may occur simultaneously with $\mathrm{nd}(F) = \Omega(n)$.

For the same example of F, we have simultaneously $\mathrm{nd}(F) = \Omega(n)$ and $\mathrm{ma}_{0,\epsilon}(F) \leq 2$ for any $\epsilon > \frac{4}{9}$, whereas by item (iii) of Theorem 5.2 the condition

$\mathrm{ma}_{0,\epsilon}(f) \leq 2$ with $\epsilon \leq \frac{1}{3}$ implies $\mathrm{nd}(f) \leq 2$. It would be interesting to close the gap $\frac{1}{3} < \epsilon \leq \frac{4}{9}$.

Proof of Theorem 5.3: The function F will be specified by its set of zeroes, that will be denoted by U. Associate with σ a constant $k = \lceil 2/\sigma^3 \rceil$. We need a set U with two properties true for sufficiently large n.

1. The complement $\{0,1\}^n \setminus U$ is not $\lfloor \alpha n \rfloor$-open for some constant $\alpha \in (0,1)$.

2. Let $u_1, \ldots, u_k$ be arbitrary pairwise distinct strings from U, and v be an arbitrary string from $\{0,1\}^k$. Define $I \subset [n]$ to be the set of positions i such that $u_1|_i u_2|_i \ldots u_k|_i = v$, where $u|_i$ stands for i-th component of u. (The sets I for different v's form the partition of $[n]$.) Then any two strings u' and u'' from $U \setminus \{u_1, \ldots, u_k\}$ agree in at most $(1+\sigma^2)|I|/2$ positions from I.

The second condition is a strengthening of the fact that U is a code with minimum distance at least $(1-\sigma^2)n/2$.

Such U exists for any $\sigma > 0$. It suffices to take $\lceil 2^{\beta n} \rceil$ strings independently at random for a constant $\beta \in (0,1)$. Property 1 holds true with high probability, provided $\alpha < \beta$. Indeed, U does not intersect an $\lfloor \alpha n \rfloor$-cylinder with probability $(1-2^{-\lfloor \alpha n \rfloor})^{\lceil 2^{\beta n} \rceil}$. So, the probability that the complement of U contains at least one $\lfloor \alpha n \rfloor$-cylinder does not exceed $2^{\alpha n}\binom{n}{\lfloor \alpha n \rfloor}(1-2^{-\alpha n})^{2^{\beta n}}$. The last value is small for $\alpha < \beta$ and large n.

Property 2 is fulfilled also with high probability. This can be easily deduced from the Chernoff bound (see Lemma 4.6), provided $\beta = \beta(\sigma)$ is small enough. (Note that the projection of U onto I consists of random strings whose length with overwhelming probability exceeds $n/2^{k+1}$.)

¿From property 1, it follows immediately that $\mathrm{nd}(F) > \alpha n$. Based on property 2, we prove the second needed condition that $\mathrm{ma}_{0,2/3+\sigma}(F) \leq 1$. This inequality can be restated as follows: for any $w \notin U$ there is a distribution $\mathbf{i}$ on $[n]$ such that for all $u \in U$ bits $u|_{\mathbf{i}}$ and $w|_{\mathbf{i}}$ coincide with probability less than $\frac{2}{3} + \sigma$. By the min-max theorem, it is equivalent to show that, given any $w \notin U$ and an arbitrary distribution $\mathbf{u}$ on U, there is an index i with

$$\mathbf{P}\left[\mathbf{u}|_i = w|_i\right] < \frac{2}{3} + \sigma. \tag{12}$$

When referring to the weight of a $u \in U$, we mean the probability that $\mathbf{u} = u$. If there is a specific u of weight at least $\frac{1}{3}$, then (12) is true for a position i where u and w differ. So we will suppose that $\mathbf{u}$ takes every its value with probability strictly less than $\frac{1}{3}$. Let us rank strings in U in descending order of their weights. Denote the weights of the first $k+1$ strings $u_1, \ldots, u_k, u_{k+1}$ by $\omega_1, \ldots, \omega_k, \omega_{k+1}$ respectively. Observe that

$$\omega_{k+1} < \frac{1}{k}. \tag{13}$$

Set $\omega = \sum_{j=1}^{k} \omega_j$. As we assume that $\omega_j < \frac{1}{3}$ for all j, there is $t \le k$ such that

$$\frac{\omega}{2} - \frac{1}{6} < \sum_{j=1}^{t} \omega_j < \frac{\omega}{2} + \frac{1}{6}. \tag{14}$$

Let $I \subset [n]$ be the set of all those positions i that

$$u_j|_i = \begin{cases} 0 & \text{for } j \le t, \\ 1 & \text{for } t < j \le k. \end{cases} \tag{15}$$

Now let $\mathbf{i}$ denote a random index from I. Our goal is to show that $\mathbf{u}|_{\mathbf{i}} = w|_{\mathbf{i}}$ with probability less than $\frac{2}{3} + \sigma$. This will imply (12) for some specific $i \in I$.

By the total probability formula

$$\mathbf{P}\Big[\mathbf{u}|_{\mathbf{i}} = w|_{\mathbf{i}}\Big] = \sum_{j=1}^{k} \mathbf{P}\Big[u_j|_{\mathbf{i}} = w|_{\mathbf{i}}\Big]\omega_j + \mathbf{P}\Big[\mathbf{u}|_{\mathbf{i}} = w|_{\mathbf{i}} \Big| \mathbf{u} \ne u_j \text{ for all } j \le k\Big](1-\omega) \tag{16}$$

We will now bound both terms in the right hand side from above. The first term is less than

$$\frac{\omega}{2} + \frac{1}{6} \tag{17}$$

by (14) and (15).

Let $m = |I|$ and $p_i = \mathbf{P}\left[\mathbf{u}|_i = w|_i \,|\, \mathbf{u} \ne u_j \text{ for all } j \le k\right]$. Without the factor of $1-\omega$, the second term in (16) can be rewritten as

$$\begin{aligned} \frac{1}{m}\sum_{i\in I} p_i &= \frac{1}{2} + \frac{1}{m}\sum_{i\in I}(p_i - \frac{1}{2}) \le \frac{1}{2} + \left(\frac{1}{m}\sum_{i\in I}(p_i - \frac{1}{2})^2\right)^{1/2} \\ &= \frac{1}{2} + \left(\frac{1}{2}\left(\frac{1}{m}\sum_{i\in I}\left(p_i^2 + (1-p_i)^2\right) - \frac{1}{2}\right)\right)^{1/2}. \end{aligned}$$

Consider two independent random strings $\mathbf{u}'$ and $\mathbf{u}''$, both having the distribution of $\mathbf{u}$ conditioned on $\mathbf{u} \ne u_j$ for all $j \le k$. Notice that the sum $\sum_{i\in I}(p_i^2 + (1-p_i)^2)$ is equal to the average number of positions where $\mathbf{u}'$ and $\mathbf{u}''$ agree. If $\mathbf{u}' \ne \mathbf{u}''$, the number of such positions does not exceed $(1+\sigma^2)m/2$ by condition 2 imposed on U at the very beginning. Therefore, the second term in (16) can be bounded by

$$\left(\frac{1}{2} + \left(\frac{\sigma^2}{2} + \mathbf{P}\left[\mathbf{u}' = \mathbf{u}''\right]\right)^{1/2}\right)(1-\omega).$$

Suppose that $\mathbf{P}\left[\mathbf{u}' = \mathbf{u}''\right] \le \sigma^2/2$. This provides us an upper bound $(1/2+\sigma)(1-\omega)$ on the second term in (16). Using also bound (17) on the first term, we obtain $\mathbf{P}\left[\mathbf{u}|_{\mathbf{i}} = w|_{\mathbf{i}}\right] < \frac{2}{3} + \sigma$, which implies (12).

If $\mathbf{P}[\mathbf{u}' = \mathbf{u}''] > \sigma^2/2$, the set $U \setminus \{u_1, \dots, u_k\}$ must contain an element whose weight exceeds $\sigma^2(1-\omega)/2$. Recall that the largest weight in this set is assigned to u_{k+1}. So, $\omega_{k+1} > \sigma^2(1-\omega)/2$. By (13) and the choice of k, we get $1-\omega < 2/(k\sigma^2) \le \sigma$. This gives us an upper bound σ on the second term in (16). Together with bound (17) on the first term, this again implies (12).

The proof is complete. ∎

Proof of Theorem 5.2

We will use items (i) and (ii) of the following lemma. Item (iii) is included there, as it complements the preceding two and shows that the same method cannot be applied to derive the conclusion $\mathrm{nd}(f) = O(1)$ from the assumption $\mathrm{ma}_{0,\epsilon}(f) \le 3$. We employ the notion of separability defined in Section 3.

Lemma 5.4 *Let f be a boolean function, w denote an arbitrary one of f, and $u_1, u_2, u_3, \dots, u_y$ denote arbitrary zeroes of f.*

(i) *if* $\mathrm{sep}(w, \{u_1, u_2\}) \le 1$ *for all* w, u_1, u_2, *then* $\mathrm{nd}(f) \le 1$;

(ii) *if* $\mathrm{sep}(w, \{u_1, u_2, u_3\}) \le 2$ *for all* w, u_1, u_2, u_3, *then* $\mathrm{nd}(f) \le 2$;

(iii) *for any y and sufficiently large n, there exists a boolean function $f : \{0,1\}^n \to \{0,1\}$ such that* $\mathrm{sep}(w, \{u_1, \dots, u_y\}) \le 3$ *for all* $w, u_1, \dots, u_y$ *but* $\mathrm{nd}(f) \ge \frac{n}{y 2^{y+1} \ln\ln n}$.

We are now able to prove Theorem 5.2. By Theorem 2.2 (ii) it suffices to prove the claims only for one-sided error. Note that the proof Theorem 3.1 (iii) gives us bound $\mathrm{sep}(w, \{u_1, \dots, u_k\}) \le \mathrm{ma}_{0,\epsilon}(f) \lceil \log k / \log(1/\epsilon) \rceil$. In particular, $\mathrm{ma}_{0,1/2}(f) \le 1$ implies $\mathrm{sep}(w, \{u_1, u_2\}) \le 1$, and $\mathrm{ma}_{0,1/3}(f) \le 2$ implies $\mathrm{sep}(w, \{u_1, u_2, u_3\}) \le 2$. A more careful inspection of the arguments shows that $\mathrm{sep}(w, \{u_1, u_2, u_3\}) \le 2$ follows also from the assumption $\mathrm{ma}_{0,2/3}(f) \le 1$. Applying claims (i) and (ii) of Lemma 5.4, we get the needed implications.

Proof of Lemma 5.4: First we introduce some unary operations over subsets of $\{0,1\}^n$ that resemble closure operators in Cantor discontinuum. Let $U \subseteq \{0,1\}^n$. Given $x \le n$, we define

$$\mathcal{C}_x(U) = \{ w \in \{0,1\}^n : \text{each } x\text{-neighborhood of } w \text{ intersects } U \}.$$

Furthermore, we define

$$\mathcal{C}_x^y(U) = \bigcup_{u_1, \dots, u_y \in U} \mathcal{C}_x(u_1, \dots, u_y).$$

Thus, $\mathcal{C}_x(U)$ consists of all strings excepting those that can be separated from U by an x-neighborhood. $\mathcal{C}_x^y(U)$ is more restricted. It contains all strings excepting those that can be separated by an x-neighborhood from any y (not necessarily distinct) elements of U.

Further on U denotes the set of zeroes of a function $f : \{0,1\}^n \to \{0,1\}$. Recall that $\mathrm{nd}(f) \le z$ iff the set of ones of f is z-open. As easily seen, the latter condition is equivalent to the equality $\mathcal{C}_z(U) = U$. It is also not hard to see that the condition $\mathrm{sep}(w, \{u_1, \ldots, u_y\}) \le x$ true for any one w and zeroes $u_1, \ldots, u_y$ of f is equivalent to $\mathcal{C}_x^y(U) = U$. Thus, the claims of the lemma can be rewritten as follows.

(i) for any $U \subseteq \{0,1\}^n$, $\mathcal{C}_1^2(U) = U$ implies $\mathcal{C}_1(U) = U$;

(ii) for any $U \subseteq \{0,1\}^n$, $\mathcal{C}_2^3(U) = U$ implies $\mathcal{C}_2(U) = U$;

(iii) for any y and sufficiently large n, there exists a set $U \subseteq \{0,1\}^n$ such that $\mathcal{C}_3^y(U) = U$ but $\mathcal{C}_z(U) \ne U$ for $z = \lfloor \frac{n}{y2^{y+1} \ln\ln n} \rfloor$.

Let us prove the first item.

Proof of (i): Suppose $\mathcal{C}_1^2(U) = U$. We have to deduce that $\mathcal{C}_1(U) = U$. This will be done if we show that U is a cylinder.

Let V be a maximum (respectively to inclusion) cylinder contained in U. We wish to show that $U = V$. Assume, to the contrary, that there is an element $u \in U \setminus V$. Let $J \subseteq [n]$ be the set of positions, whose entries define V. Let $I \subset J$ be the subset of positions, where u has the same entries. We choose $v \in V$ so that v and u differ at all positions outside J. Thus, v and u coincide only at positions from I. Notice that $\mathcal{C}_1(u, v)$ is a cylinder definable by setting the coordinates from I as in u (or v). As this cylinder properly contains the cylinder V and is itself contained in $\mathcal{C}_1^2(U) = U$, we get a contradiction. This completes the proof of item (i). ▮

Proof of (ii): We will use the following simple fact. Given three boolean vectors $u_1, u_2, u_3 \in \{0,1\}^n$, let $\mathrm{MAJ}(u_1, u_2, u_3)$ be a vector, whose i-th entry occurs at least twice among i-th entries of u_1, u_2, u_3.

Claim 1: If $u_1, u_2, u_3 \in V$, then $\mathrm{MAJ}(u_1, u_2, u_3) \in \mathcal{C}_2^3(V)$.

Proof: As easily seen, $\mathrm{MAJ}(u_1, u_2, u_3) \in \mathcal{C}_2(u_1, u_2, u_3)$. □

Let us turn to claim (ii). Consider $U \subseteq \{0,1\}^n$ such that $\mathcal{C}_2^3(U) = U$. We have to prove $\mathcal{C}_2(U) = U$. Suppose $u \in \mathcal{C}_2(U)$ and deduce $u \in U$. It is not hard to see that the closure operators $\mathcal{C}$ commutate with shifting by any element of $\{0,1\}^n$. So, without loss of generality we may assume $u = 1^n (= 11\ldots1)$.

We say that $V \subseteq \{0,1\}^n$ has a *complete i-shadow* if for any $I \subseteq [n]$, $|I| = i$, some $v \in V$ has 1 at all positions from I.

Claim 2: Let $0 \le i \le n-2$. Then U has a complete $(i+2)$-shadow.

Proof: We proceed by induction on i. In the case $i = 0$ the claim is a reformulation of the fact that $1^n \in \mathcal{C}_2(U)$. Suppose the claim is true in the case of $i-1$. Look at the case of i, where $i \ge 1$. Choose an arbitrary set of positions $I \subseteq [n]$ with $|I| = i+2$. We have to show that U contains some v with 1's on I. Pick three distinct positions $i_1, i_2, i_3 \in I$. Let $I_s = I \setminus \{i_s\}$, $s = 1, 2, 3$. By the induction hypothesis, U contains some vectors u_1, u_2, u_3 with 1's on I_1, I_2, I_3, respectively. By Claim 1, $\mathrm{MAJ}(u_1, u_2, u_3)$ is in $\mathcal{C}_2^3(U)$ and, therefore, in U by our assumption. It is easy to see that $\mathrm{MAJ}(u_1, u_2, u_3)$ has 1's at all positions in I. $\square$

When $i = n - 2$, Claim 2 means that $1^n \in U$, completing the proof of claim (ii).

∎

Note that claim (i) can be proven similarly to claim (ii). It suffices to replace $\mathrm{MAJ}(u_1, u_2, u_3)$ with $\mathrm{OR}(u_1, u_2)$.

Proof of (iii): Given $V \subseteq \{0,1\}^n$ whose elements $v_1, \ldots, v_y$ are arbitrarily ordered, we denote a matrix of size y by n with rows $v_1, \ldots, v_y$ by $M(V)$. By N we denote the matrix of size y by $y - 1$

$$\begin{matrix} 0 & 0 & \ldots & 0 \\ 1 & 0 & \ldots & 0 \\ 1 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \ldots & 1 \end{matrix}$$

Claim 3: Let $V \subseteq \{0,1\}^n$. If the matrix $M(V)$ contains all the columns of the matrix N, then $\mathcal{C}_3(V) = V$.

Proof: Without loss of generality, suppose that the first $(y-1)$ columns of $M(V)$ make up the matrix N. Consider a vector v in $\mathcal{C}_3(V)$. We have to show that v must be in V. Let m be the first position, where v has 0 (let $m = n + 1$ if $v = 1^n$). The condition $v \in \mathcal{C}_3(V)$ means that for any $I \subseteq [n]$ with $|I| = 3$, some v_j coincides with v on I. For this reason, $v = v_y$ if $m \geq y$, and $v = v_m$ otherwise. $\square$

Given y and n, we set $l = \lfloor z 2^{z+1} \ln n \rfloor$ for $z = \lfloor \frac{n}{y 2^{y+1} \ln \ln n} \rfloor$. Choose $u_1, \ldots, u_l$ from $\{0,1\}^n$ randomly and independently from each other, and put $U = \{u_1, \ldots, u_l\}$. We are going to show for n large enough that three events $1^n \in \mathcal{C}_z(U)$, $1^n \notin U$, and $\mathcal{C}_3^y(U) = U$ simultaneously take place with non-zero probability. This will imply what we need. Let us show that every one of the three events above does not occur with small probability.

$1^n \notin \mathcal{C}_z(U)$ means that for some $I \subseteq [n]$ with $|I| = z$ none of the $u_1, \ldots, u_l$ has all 1's on I. This happens with probability at most $\binom{n}{z}(1 - 2^{-z})^l \leq n^z \exp\{-l 2^{-z}\} \leq n^{-z}$, which is less than $\frac{1}{3}$ for z large enough. It follows $\mathbf{P}[1^n \notin \mathcal{C}_z(U)] < \frac{1}{3}$ for z large enough.

$\mathbf{P}[1^n \in U] \leq l 2^{-n} < \frac{1}{3}$ for n large enough.

$\mathcal{C}_3^y(U) \neq U$ implies that for some $V \subseteq U$ with $|V| = y$, V is properly contained in $\mathcal{C}_3(V)$. So, $\mathbf{P}[\mathcal{C}_3^y(U) \neq U] \leq \binom{l}{y} \mathbf{P}[\mathcal{C}_3(V) \neq V]$, where $V = \{v_1, \ldots, v_y\}$ consists of random elements of $\{0,1\}^n$. By Claim 3, $\mathcal{C}_3(V) \neq V$ implies that the matrix $M(V)$

does not have at least one of the columns of the matrix N. Hence,

$$\mathbf{P}\left[\mathcal{C}_3^y(U) \neq U\right] \leq \binom{l}{y} y(1-2^{-y})^n \leq l^y y \exp\left\{-n2^{-y}\right\} < \frac{1}{3}$$

for n large enough.

This proves claim (iii). ▮

The proof of Lemma 5.4 is complete. ■

6 Conclusion and open problems

The main question we leave open is if $\mathrm{am}_\epsilon(f)$ and $\mathrm{nd}(f)$ are polynomially related. A variation of this question can be if $\mathrm{am}_\epsilon^{(\mathrm{polylog}\, n)}(f) = \mathrm{polylog}\, n$ or $\mathrm{ma}_\epsilon^{(\mathrm{polylog}\, n)}(f) = \mathrm{polylog}\, n$ implies $\mathrm{nd}(f) = \mathrm{polylog}\, n$. For F, the check function of an αn-uniform code with minimum distance δn, we have shown that $\mathrm{ma}_{0,1-\delta/2}(F) = 1$, while $\mathrm{nd}(F) = \Omega(n)$ and $\mathrm{am}_\epsilon(F) = \Omega(\log n)$. It would be insightful to improve our logarithmic lower bound on $\mathrm{am}_\epsilon(F)$ or, alternatively, give an upper bound.

Another related question is if $\mathrm{am}_\epsilon(f) = O(1)$ implies $\mathrm{nd}(f) = O(1)$. We answer it in affirmative only in the particular case of $\mathrm{am}_\epsilon(f) \leq 1$. Moreover, we prove that $\mathrm{ma}_{0,2/3}(f) \leq 1$ implies $\mathrm{nd}(f) \leq 2$. The error $\frac{2}{3}$ is here the exact threshold, as one can achieve simultaneously $\mathrm{nd}(F) = \Omega(n)$ and $\mathrm{ma}_{0,\epsilon}(F) = 1$ for arbitrary $\epsilon > \frac{2}{3}$. We prove that $\mathrm{ma}_{0,1/3}(f) \leq 2$ also implies $\mathrm{nd}(f) \leq 2$, whereas in the aforementioned example $\mathrm{nd}(F) = \Omega(n)$ and $\mathrm{ma}_{0,\epsilon}(F) \leq 2$ for arbitrary $\epsilon > \frac{4}{9}$. It would be interesting to investigate the range $\frac{1}{3} < \epsilon \leq \frac{4}{9}$. Does there exist the theshold in this case too?

Given a boolean function f and an integer d, denote $\mathrm{err}_d(f) = \inf\left\{\epsilon : \mathrm{ma}_{0,\epsilon}(f) \leq d\right\}$ (for convenience let $\inf \emptyset = 1$). Clearly, $\mathrm{err}_2(f) \leq \mathrm{err}_1(f)^2$. It is interesting to give an example when the latter inequality is strict. Is this true for the above code-check function F?

Acknowledgments

We thank Alexander Barg and Alexander Razborov for useful discussions.

References

[1] L. Babai. Trading group theory for randomness. In *Proc. of the 17th ACM Ann. Symp. on the Theory of Computing (STOC)*, pages 421–429, 1985.

[2] M. Blum and R. Impagliazzo. Generic oracles and oracle classes. In *Proc. of the 28th IEEE Ann. Symp. on Foundations of Computer Science (FOCS)*, pages 118–126, 1987.

[3] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Math. Stat.*, 23:493–509, 1952.

[4] O. Goldreich, Y. Mansour, and M. Sipser. Interactive proof systems: provers that never fail and random selection. In *Proc. of the* 28*th IEEE Ann. Symp. on Foundations of Computer Science (FOCS)*, pages 449–461, 1987.

[5] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989.

[6] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof systems. In *Proc. of the* 18*th ACM Ann. Symp. on the Theory of Computing (STOC)*, pages 59–68, 1986.

[7] J. Hartmanis and L. Hemachandra. Robust machines accept easy sets. *Theoretical Computer Science*, 74(2):217–226, 1990.

[8] R. Impagliazzo and M. Naor. Decision trees and downward closures. In *Proc. of the* 3*rd ACM Ann. Conf. on Structure in Complexity Theory*, 1988.

[9] S. Jukna and A. Razborov. Neither reading few bits twice nor reading illegally helps much. *Electronic Colloquium on Computational Complexity,* TR96-037, 1996.

[10] M. L. Minsky and S. A. Papert. *Perceptrons.* MIT Press, Cambridge, MA, second edition, 1988.

[11] A. A. Muchnik and N. K. Vereshchagin. A general method to construct oracles realizing given relationships between complexity classes. *Theoretical Computer Science*, 157:227–258, 1996.

[12] N. Nisan. CREW PRAMs and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991.

[13] W. Scharlau. Selbstduale Goppa-codes. *Mathematische Nachrichten*, 143:119–122, 1989.

[14] G. Tardos. Query complexity, or why it is difficult to separate $NP^A \cap coNP^A$ from P^A by a random oracle. *Combinatorica*, 9:385–392, 1989.

[15] N. Vereshchagin. Complexity of computation on functions and relativized complexity of computation on words. *Manuscript*, 1989.

[16] N. Vereshchagin. Relativizable and nonrelativizable theorems in the polynomial theory of algorithms. *Russian Acad. Sci. Izv. Math.*, 42(2):261–298, 1994.

[17] A. Yao. Probabilistic computations: towards a unified measure of complexity. In *Proc. of the* 18*th IEEE Ann. Symp. on Foundations of Computer Science (FOCS)*, pages 222–227, 1977.

[18] S. Zachos and M. Fürer. Probabilistic quantifiers vs. distrustful adversaries. In *Proc. Foundations of Software Technology and Theoretical Computer Science*, pages 443–455. Springer-Verlag, 1987. LNCS v. 287.